

50 听起来很恐怖的数据库幻读，到底是个什么奇葩问题？

上一讲我们给大家讲解了不可重复读这个问题，这个问题简单来说，就是一个事务多次查询一条数据，结果每次读到的值都不一样，这个过程中可能别的事务会修改这条数据的值，而且修改值之后事务都提交了，结果导致人家每次查到的值都不一样，都查到了提交事务修改过的值，这就是所谓的不可重复读。

不可重复读，就是一条数据的值没法满足多次重复读值都一样，别的事务修改了值提交之后，就不可重复读了，说着有点拗口，不过相信大家应该反复看两遍，应该都能理解

包括之前的脏写和脏读其实也都分别代表了不同的数据库的问题，脏写就是两个事务没提交的状况下，都修改同一条数据，结果一个事务回滚了，把另外一个事务修改的值也给撤销了，所谓脏写就是两个事务没提交状态下修改同一个值。

脏读就是一个事务修改了一条数据的值，结果还没提交呢，另外一个事务就读到了你修改的值，然后你回滚了，人家事务再次读，就读不到了，也就是说人家事务读到了你修改之后还没提交的值，这就是脏读了。

而不可重复读，针对的是已经提交的事务修改的值，被你事务给读到了，你事务内多次查询，多次读到的是别的已经提交的事务修改过的值，这就导致不可重复读了。

这个数据库的多种并发问题，确实很拗口，需要大家好好理解。

今天我们来最后讲一种数据库的并发问题，就是听着有点恐怖的幻读问题，幻读听起来很恐怖，搞的跟邪恶的巫师给你搞什么魔法一样，是不是？

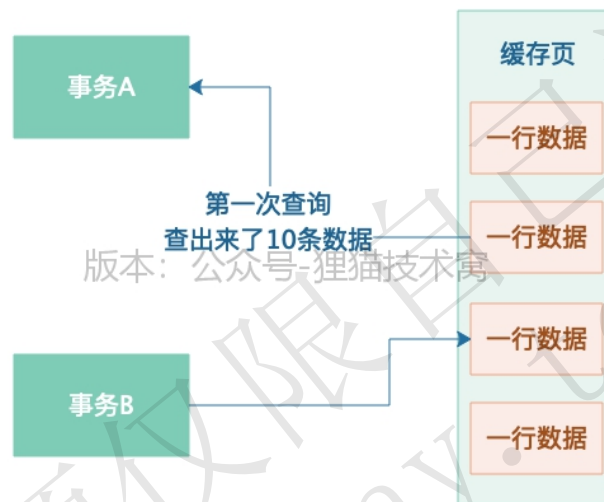
其实没那么恐怖的，我们今天给大家来讲讲。

简单来说，你一个事务A，先发送一条SQL语句，里面有一个条件，要查询一批数据出来，比如“select * from table where id>10”，类似这种SQL

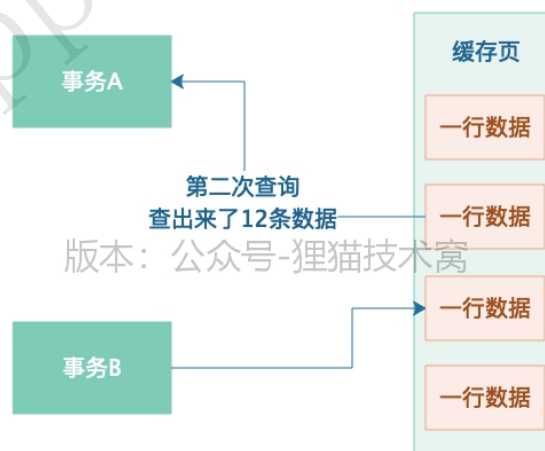
然后呢，他一开始查询出来了10条数据，如下图所示。



接着这个时候，别的事务B往表里插入了几条数据，而且事务B还提交了，如下图所示，此时多了几行数据出来。



接着事务A此时第三次查询，再次按照之前的一模一样的条件执行“select * from table where id>10”这条SQL语句，由于其他事务插入了几条数据，导致这次他查询出来了12条数据，如下图所示。



于是此时事务A开始怀疑自己的双眼了，为什么一模一样的SQL语句，第一次查询是10条数据，第二次查询是12条数据？难道刚才出现了幻觉？导致我刚才幻读了？这就是幻读这个名词的由来。

幻读指的就是你一个事务用一样的SQL多次查询，结果每次查询都会发现查到了一些之前没看到过的数据

注意，幻读特指的是你查询到了之前查询没看到过的数据！此时就说你是幻读了。

说实在的，大家看完最近几篇文章，应该都有一个感觉，就是脏写、脏读、不可重复读、幻读，都是因为业务系统会多线程并发执行，每个线程可能都会开启一个事务，每个事务都会执行增删改查操作。

然后数据库会并发执行多个事务，多个事务可能会并发的对缓存页里的同一批数据进行增删改查操作，于是这个并发增删改查同一批数据的问题，可能会导致我们说的脏写、脏读、不可重复读、幻读，这些问题。

所以这些问题的本质，都是数据库的多事务并发问题，那么为了解决多事务并发问题，数据库才设计了事务隔离机制、MVCC多版本隔离机制、锁机制，用一整套机制来解决多事务并发问题，接下来，我们将要深入讲解这些机制，让大家彻底能够理解数据库内部的执行原理。

深刻理解了原理之后，后续我们讲解各种数据库优化实践案例，大家才能深刻理解，就跟我之前的《从0开始带你成为JVM实战高手》一样，先得透彻理解了JVM运行的工作原理，才能理解如何对JVM进行实战优化。

只不过JVM的内部运行原理比数据库而言相对内容少一些，所以当时很快就进入了大量的实践案例阶段，而数据库的话，内部原理会更加复杂一些，所以耗费的时间会多一些。

End